Journal Homepage: -www.journalijar.com

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH (IJAR)

Article DOI:10.21474/IJAR01/18003
DOI URL: http://dx.doi.org/10.21474/IJAR01/18003

ISSN NO. 2320-5407

RESEARCH ARTICLE

PREDICTIVE SIMULATION OF THE ELECTRICAL PRODUCTION OF A PHOTOVOLTAIC INSTALLATION BY ARTIFICIAL NEURAL NETWORKS

Mamadou Salif Diallo[1,2], Abdoul Aziz Cissé[2] and Hamet Yoro BA[3]

1. Energy Efficiency and Systems Research Group, Université Alioune Diop, Bambey Sénégal.
2. Laboratoire des Sciences et Techniques et de l'Environnement, Ecole Polytechnique de Thiès, Sénégal.
3. Ecole Polytechnique de Thiès BP A10 Thiès, Département Génie Electromécanique, Sénégal.

Manuscript Info

Manuscript History
Received: 15 October 2023
Final Accepted: 18 November 2023
Published: December 2023

Key words:-
Prediction, Artificial Neural Networks (ANNs), Photovoltaic Installation, Multi-Variable Regression, Prosumer

Abstract

This paper explores the possibility of predicting the production of a prosumer by artificial neural networks. We made a cross comparison of two ANNs architectures (looped and unlooped) with respect to multivariable regression in order to have an efficient and reliable tool for predicting the production of a photovoltaic installation from meteorological data (solar irradiance and ambient temperature).To accomplish these goals, we used monitoring data of an installation over a period of **72** days to realize, train and test two ANNs topologies (looped and unlooped) which are trained with the Levenberg-Marquardt algorithm. After training and testing, it first appears that the neural networks show the best performance compared to the multivariable regression method. Then, among the two architectures, the one with the lowest uncertainties ($MAPE$=24.489%; $RMSE$=436.08 and MBE=30.93) is the feed-forward architecture.In this article, we have shown the importance of managing energy supply and demand, but also the main drawbacks of the current electricity network in Senegal.In summary, we have developed a model capable for predicting the production of a prosumer based on meteorological data (sunshine and ambient temperature).

Introduction:-

With the evolution of electricity networks and the emergence of renewable energies, the centralized structure of current networks has shown its weaknesses [1]. Indeed, the philosophy of modern electrical networks is giving rise to the concept of smart network. Thus, we leave a centralized structure for a distributed structure where the flow of information is bidirectional. In this situation, consumers become producers, they own one or more ways of production, but at the same time they use the energy of the network in case of a deficit, we refer to as self-producers or prosumers[2].

Thus, in a context where the excess energy production is traded on the energy market, the prosumer must secure a minimum power requirement in order to have an electricity purchase agreement. This amounts to controlling his production over time.

Corresponding Author:- Mamadou Salif Diallo
Address:- Energy Efficiency and Systems Research Group, Université Alioune Diop, Bambey Sénégal.

A prosumer may not directly act on the quantity that it consumes, nor on the quantity produced. Moreover, we know that in self-consumption, the overflow of energy potentially injected into the network is only the difference between the quantity produced and the quantity consumed. Therefore, it would be difficult to really control the energy overflow, hence the necessity of an efficient prediction.

In order to carry out the prediction, a lot of research work has proven the efficiency of artificial neural networks applied to energy systems [3][4][5][6].

The aim of this work is to choose a more efficient neural architecture for the prediction of the energy overflow of a prosumer.

Materials andMethods:

- Description of the proposed photovoltaic system

The experimental data base was provided by a photovoltaic installation measuring 3.5 kWp.

The energy produced by the photovoltaic modules depends to a considerable part on the geographical position and the sunshine duration of the location[4]. This statement can be written:

$$E(t) = P(t) * S \qquad (0\text{-}1)$$

With $P(t)$the instantaneous power and S is the day length obtained with the following expression:

$$S = \frac{2}{15}\cos^{-1}(-\tan L \tan \delta) \qquad (0\text{-}2)$$

Ldenotes the latitude of the place and **δ**is the angle of declination, i.e. the angle formed by the direction of the sun and the earth's equatorial plane.

The collected data concern the periodin the current pandemic covid 19 from **18/04/2020** to **28/06/2020**. The samples are distributed over a summer period of 3 months (April, May, June). We chose this period because of data availability constraints.

Thus, we have modeled the consumption and production curves from the monthly averages obtained for atypical day of each month.

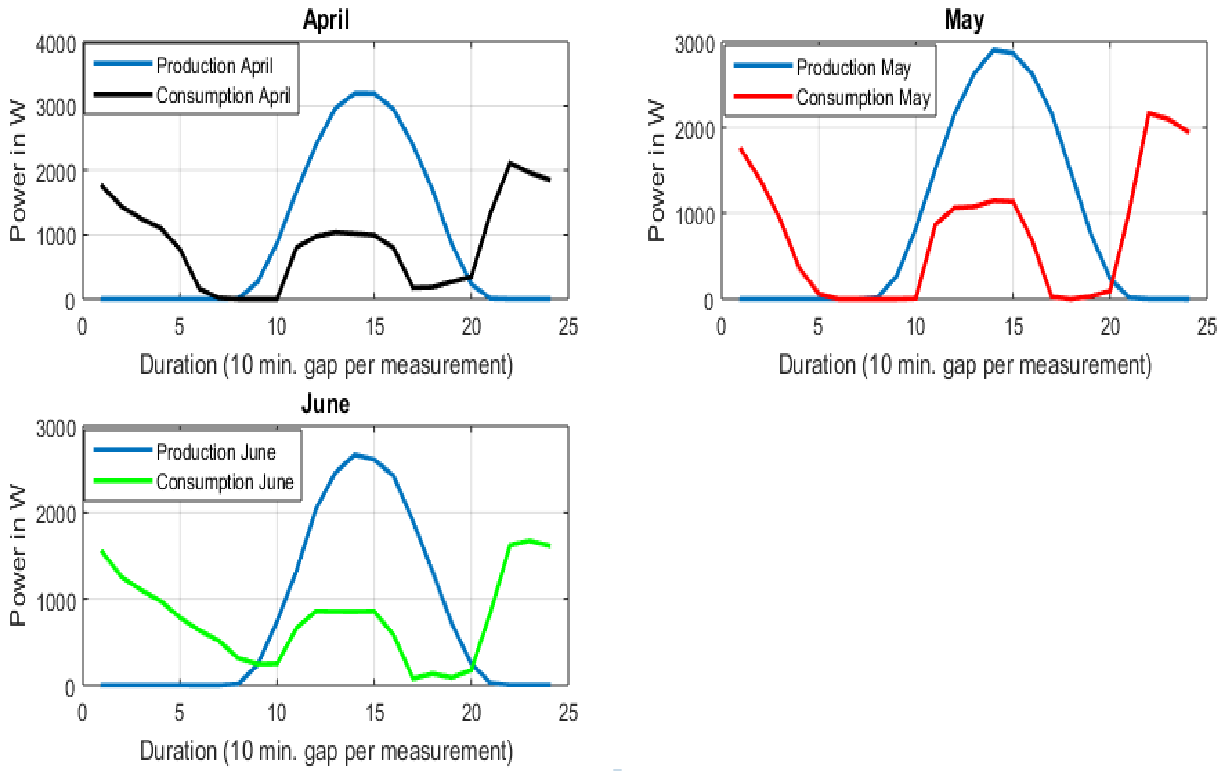


Figure 0-1:- Consumption and production output profile of the photovoltaic installation.

These three graphs indicate that for this prosumer the evolution of production is contrary to that of consumption. Consequently, in the intervals where there is an energy deficit, the self-producer pulls its consumption from the network. This phenomenon of compensation could bring him to not really benefit from his installation.

It is therefore important to be able to predict its production in order to optimize the minimum power to be guaranteed to the network manager in order to have an electricity purchase contract.

Presentation of the prediction models used
In the literature there are several prediction models. In this article, we propose to work on two methods: regression and artificial neural networks.

Regression is a statistical process used to study multifactorial data. It is one of the most famous techniques for predicting and expressing relationships between variables of interest (dependent and independent variables)[7].

The simplest regression model is represented by a simple linear regression model which is a model with a single explanatory variable (x) having a relationship with the response (y) in a simple straight line, as shown below:

$$y = \beta_0 + \beta_1 x \qquad (0\text{-}3)$$

Where β_0 the y-intercept and β_1 the steering coefficient of the line.

Moreover, if we want to take into account simultaneously the influence of several explanatory variables on a given response variable, we will turn to another form of regression analysis called multiple or multivariable regression. Its fundamental model is of the form:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \qquad (0\text{-}4)$$

With β_1 and β_2 the regression coefficients and β_0 the y-intercept.

Thus, a multiple regression model can be developed to predict the output power of a photovoltaic array using meteorological parameters such as ambient temperature (T) and solar radiation (G). There are many methods to find the coefficient of the regression model. In this paper we will use empirical methods, based on MATLAB.

Artificial neural networks
A neural network is an assembly of interconnected elementary constituents called « neurons », each of which performs a simple processing but whose interaction as a whole gives rise to complex global properties[3]. Indeed, a neuron is a non-linear and bounded algebraic function whose value depends on parameters called coefficients or weights[8].

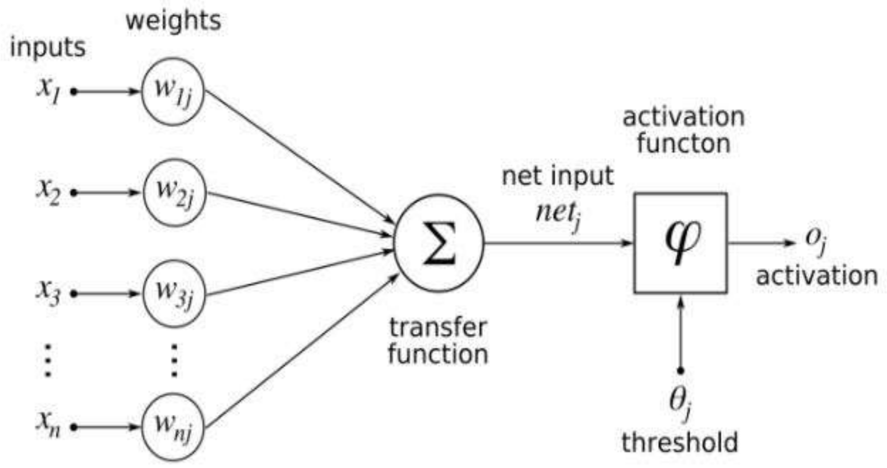


Figure 0-2:- Diagram of a neuron.

We can see that a neuron is essentially made up of an integrator that performs the weighted sum of its inputs. The result s of this sum is then transformed by a transfer function f which produces the output y of the neuron. The

$n$inputs of the neuron correspond to the vector $x = [x_1, x_2, \dots x_n]^T$while$w = [w_1, w_2, \dots w_n]^T$represents the neuron weight vector. Thus, the output s of the integrator is given by the following relation:

$$s = \sum_{i=1}^{n} w_{ij}\, x_i \pm b = w_{11}x_1 + w_{21}x_2 + \cdots + w_{n1}x_n \pm b \qquad (0\text{-}5)$$

With$x_i$the input vectors, $w_{ij}$the weights of the neuron connections, $(b)$the bias or activation threshold (input that often takes the values -1 or +1).

The sum$s$, also called activation level, is then subjected to an activation function f. When $s$reaches or exceeds the threshold$b$then the argument of $f$becomes positive (or null). Otherwise it is zero. There are several activation functions. But the sigmoid function is most often used, whose form is close to a step. Thus, the answer $y$for this function is given by:

$$y = \frac{1}{1 + exp^{-s}} \qquad (0\text{-}6)$$

Thus, each neuron functions independently of the others in such a way that the whole forms a massively parallel system. The information is stored in a distributed manner in the network in the form of synaptic coefficients or activation functions, so there is no memory area and no calculation area, both are intimately linked.

The architecture of a neural network depends on the task to be learned. A neural network is usually composed of several layers of neurons, from inputs to outputs.

There are two main types of neural network architectures: unlooped neural networks and looped neural networks.

Unlooped neural networks are static objects: if the inputs are independent of time, so are the outputs. They are mainly used to perform non-linear function approximation tasks, classification or modeling of non-linear static processes. While looped neural networks are dynamic systems, governed by differential equations, and can have any topology with loops bringing back to the inputs the value of one or more outputs. Looped neural networks are used to perform dynamic system modeling, process control, or filtering tasks.

The mechanism by which the network parameters are adjusted characterizes the network learning algorithm. In prediction, network learning is usually done in a supervised manner, meaning that the network is trained by providing its desired inputs and outputs. Supervised learning allows to establish a relationship between the training data. Then this relationship is used on active data to predict what will happen.

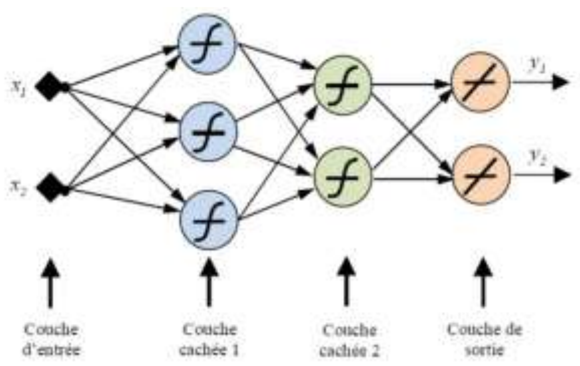
On MATLAB, the default learning algorithm is the Levenberg-Marquardt (LM) algorithm. This algorithm calculates the update of network weights using second derivatives based on Newton's method [5][9]. The LM algorithm is less complex than the gradient method, since the computation is done by Jacobian matrix instead of Hessian matrix. The updating of the weight is done by the following iterative function:

$$w_{k+1} = w_k - [J^T J + \mu J]^{-1} J^T e \qquad (0\text{-}7)$$

Where$J$the Jacobian matrix that contains the first derivative of the error, $\mu$is the Marquardt parameter that must be updated according to thedecreasing rate of output and et$e$the effective error.

Ifμgoes around zero we are approaching Newton's method,if it tends to infinite we are moving to the method of gradient descent with small steps as[9].

Figure 0-3:- Unlooped network (left) et looped network (right).



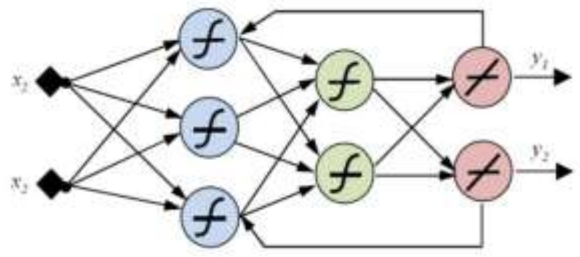


Figure 0-4:- Unlooped network (left) et looped network (right)

To estimate the performance of a neural network, statistical errors are usually used. Among these errors we have the root mean square error (RMSE) and the mean absolute error (MAE or MAPE in percentage). The RMSE represents the measure of the variation of the predicted values around the measured values. In addition, RMSE indicates the performance of the model used to predict future values. If it is positive and large enough it means that there is a large deviation of the predicted value from the measured value. It can be calculated by the following equation:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(P_i - M_i)^2} \qquad (0\text{-}8)$$

Where$P_i$is the predicted valueand$M_i$the measured value.

It should be noticed that the calculation of the RMSE for ANNs is different for regression. Indeed, for a multiple regression we divide by the degree of freedom $\boldsymbol{n}$-$\boldsymbol{k}$-$\mathbf{1}$ with k the number of explanatory variables.

MAPE can highlight the overall accuracy of a neural network. Thus, decreasing MAPE simply means improving the average prediction. MAPE can be calculated by the following equation:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{M-P}{M}\right| \qquad (0\text{-}9)$$

The use of MAPE is different from that of RMSE. MAPE is less sensitive to large errors, whereas RMSE penalizes large deviations significantly by squaring the error.

Simulation and selection of the appropriate forecasting model

Simulation Methodology:-

First of all, we partitioned the data set into three distinct sets: the learning set (70%), the validation set (15%) and the test set (15%).

First, we performed a simulation with the multivariable regression model using the learning set from our database. Thus,as doing in [10]the regression model allowed us to extract a correlation between the input values (solar irradiation and ambient temperature) and the output values (power delivered). This relation is as shown below:

$$S_{reg} = -969{,}06 + (5{,}679 * T_{amb}) + (3{,}7814 * G) \quad (0\text{-}1)$$

whereGis the solar irradiationandT_{amb} the ambient temperature.

So, this relationship will be used by the algorithm just on the input data of the test set to see the answer given by the multivariate regression.

Then using a few lines of code on MATLAB we built an unlooped ANN architecture called feed-forward and another looped architecture called cascade-forward. Each architecture was tested on the learning set to train the network, then the ability of the trained network to generalize was monitored on the validation set, and finally each architecture was subjected to the test set to verify their ability to forecast production.

The learning algorithm used is the Levenberg-Marquardt algorithm and we have set the maximum number of iterations to 200 and the maximum learning time to 60 seconds.

For the choice of the neurons of the hidden layer we have chosen 5 for both topologies after several simulations. Indeed, there is no appropriate procedure to choose the optimal number of neurons in the hidden layer. The choice factors depend on the number of input and output units, the number of learning cases, the complexity of the error function, the network architecture and the learning algorithm [5]. So, we can state that the best approach to find the optimal number of hidden neurons is by trial and error. We have tested several times and compared the errors.

After several tries, these are the two ANNs architectures designed:
In order to choose the best model, we will test the models on the input data of the test set and then compare the statistical errors (RMSE and MAPE) of the different models used.

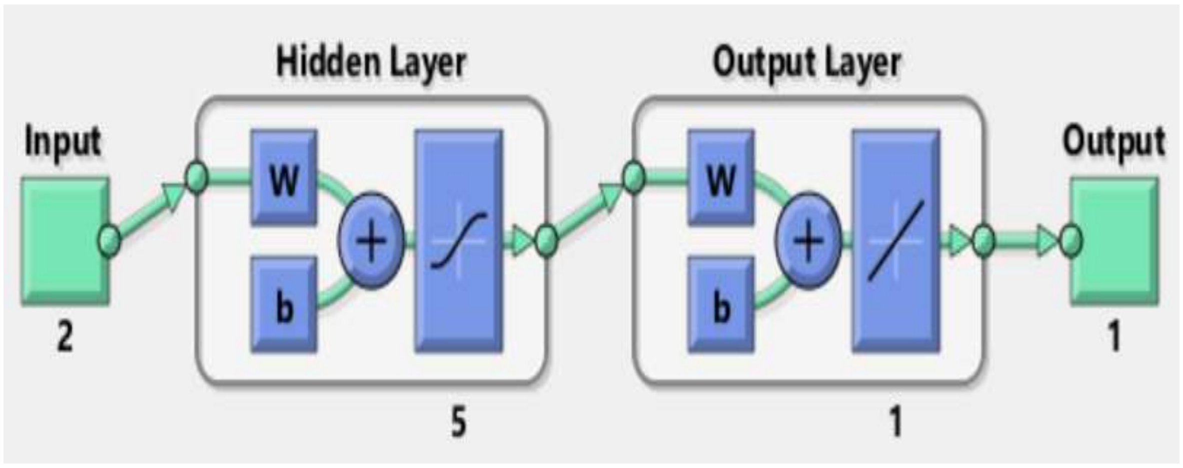


Figure 0-1:- Feed-forward architecture (unlooped) with a single hidden layer of 5 neurons (Source: MATLAB).

Reminder that the test set is 15% of the total data, which is equivalent to 1556 measurements or 11 days.

.

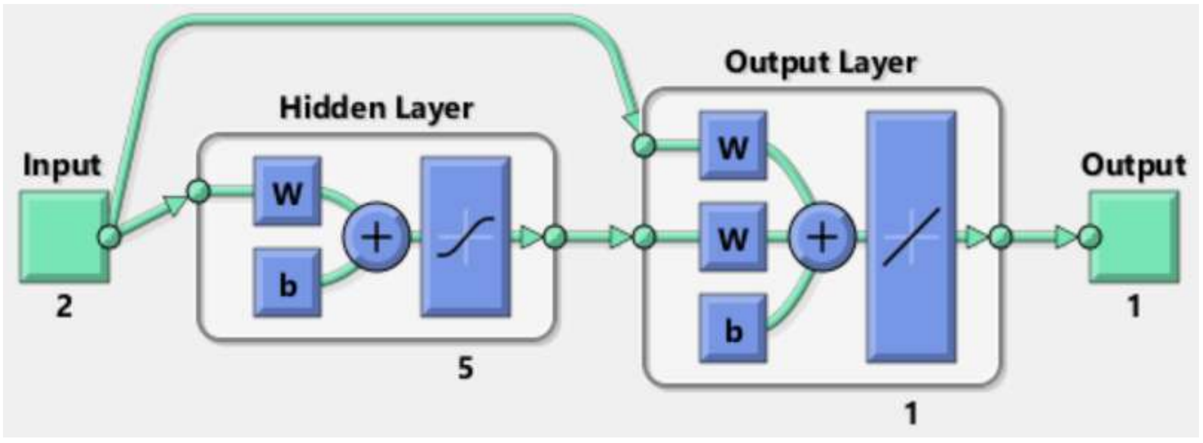


Figure 0-1:- Cascade-forward architecture (looped) with a single hidden layer of 5 neurons (Source: MATLAB).

Results and Discussion:-

The Neural Networks Toolbox of the MATLAB software was used to train, validate and test the two architectures thus developed. After creating the ANNs and running several simulations, we compared each ANN architecture with the multivariable regression.

Feed-forward and multivariableregression

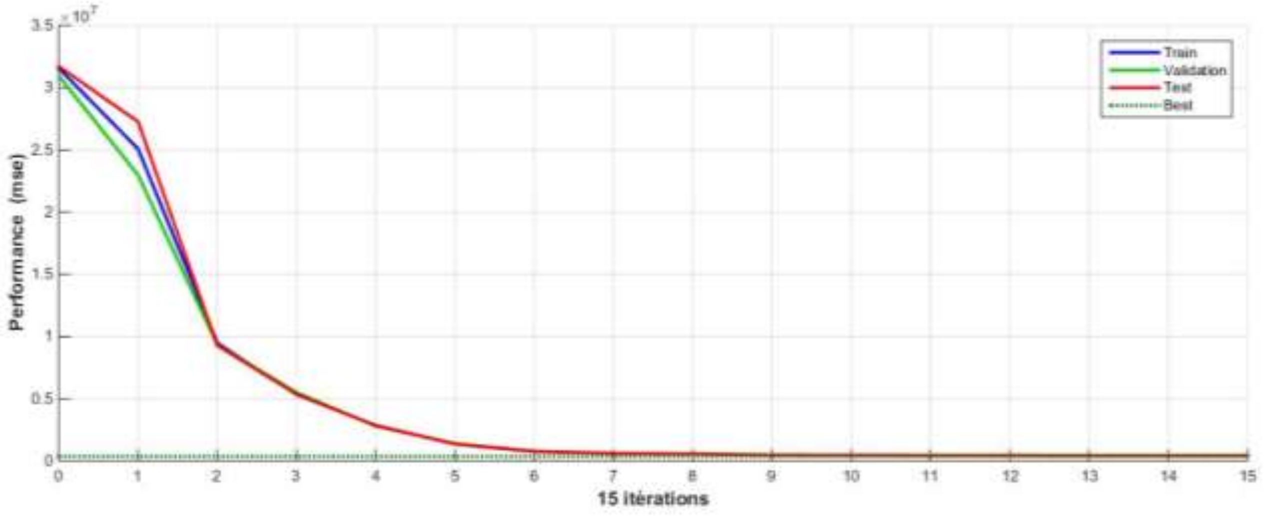


Figure IV-1:- Evolution of performance during the learning process.

After simulating several times, we find that the training stops at an MSE limit of 0.001 predefined by MATLAB. Thus, the network parameters (weight and bias) selected correspond to the minimum limit of the validation error. Already at 8 iterations the best performance is reached as shown in the figure below.

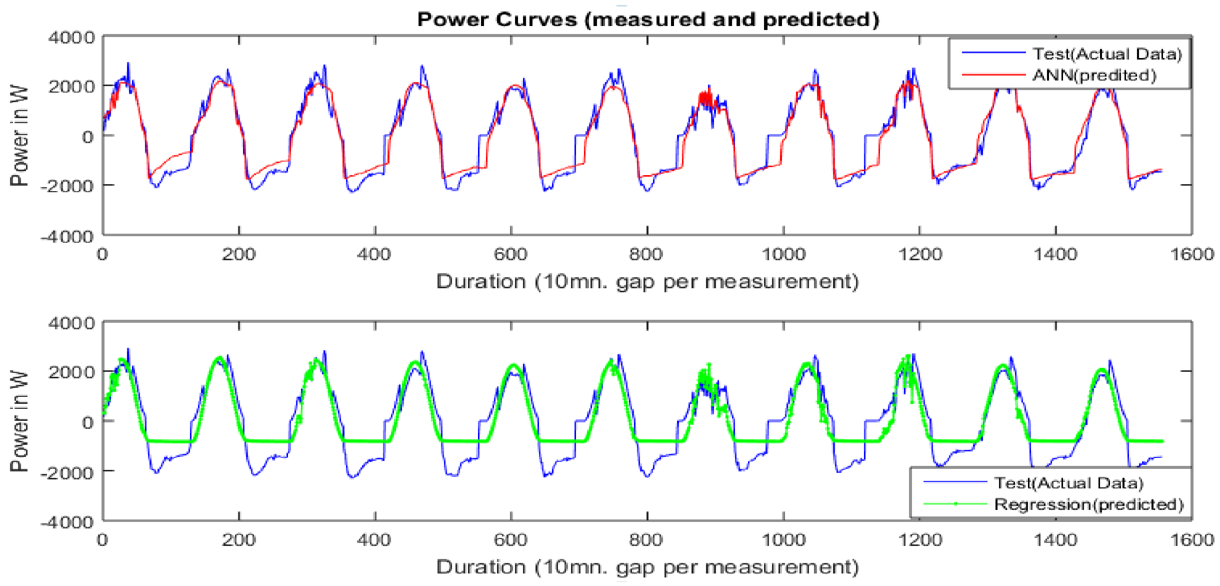


Figure IV-2:- Result of the simulation with feed-forward architecture (top) compared to regression (bottom).

Compared to multivariable regression the feed-forward architecture gives the following results:

Tableau IV-1:- Error table (feed-forward and regression).

	MAE	MAPE (%)	RMSE
ANN (feed-forward)	326.17	24.489	436.08
Multivariable regression	655.11	49.186	738.

Figures IV-1 and IV-2 as well as the error table clearly show us that the data predicted by the feed-forward architecture are closer to the target data compared to the data predicted by the regression.

Cascade-forward architectureand Multivariable regression

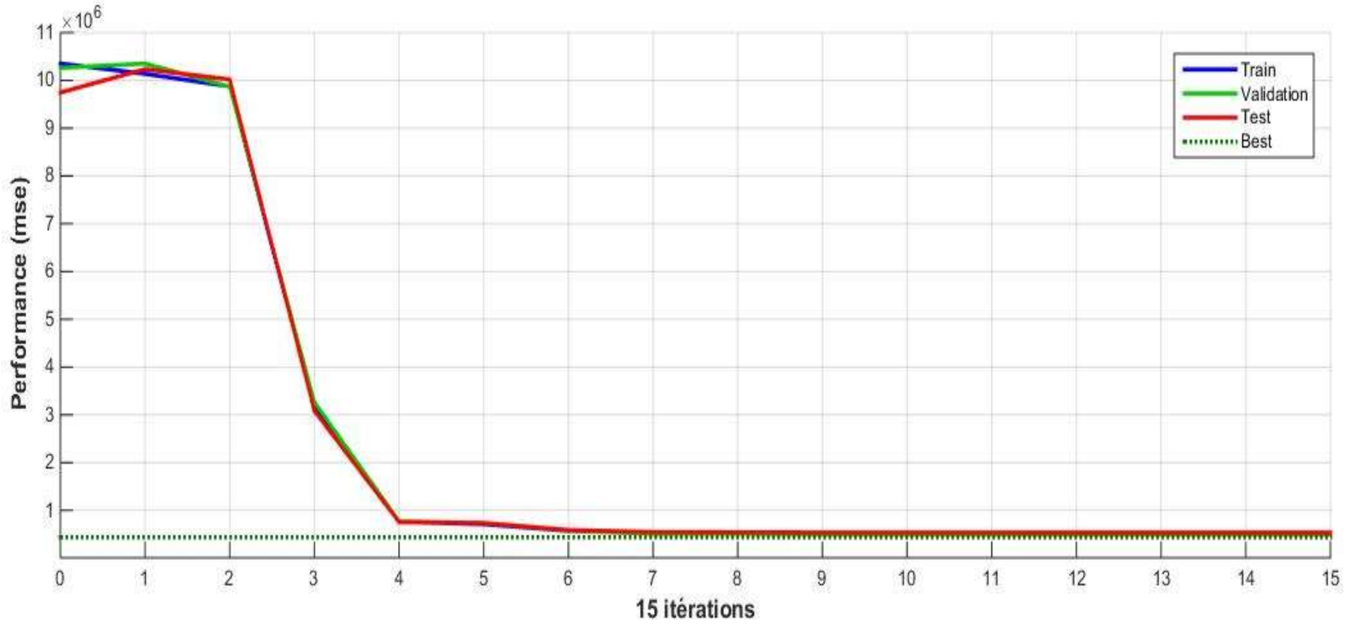


Figure IV-3:- Evolution of the performance during the learning phase (cascade-forward architecture).

After simulation we always notice that at 8 iterations, we get the best performance.

The prediction results for the cascade-forward architecture compared to multivariable regression are shown on the graph below:

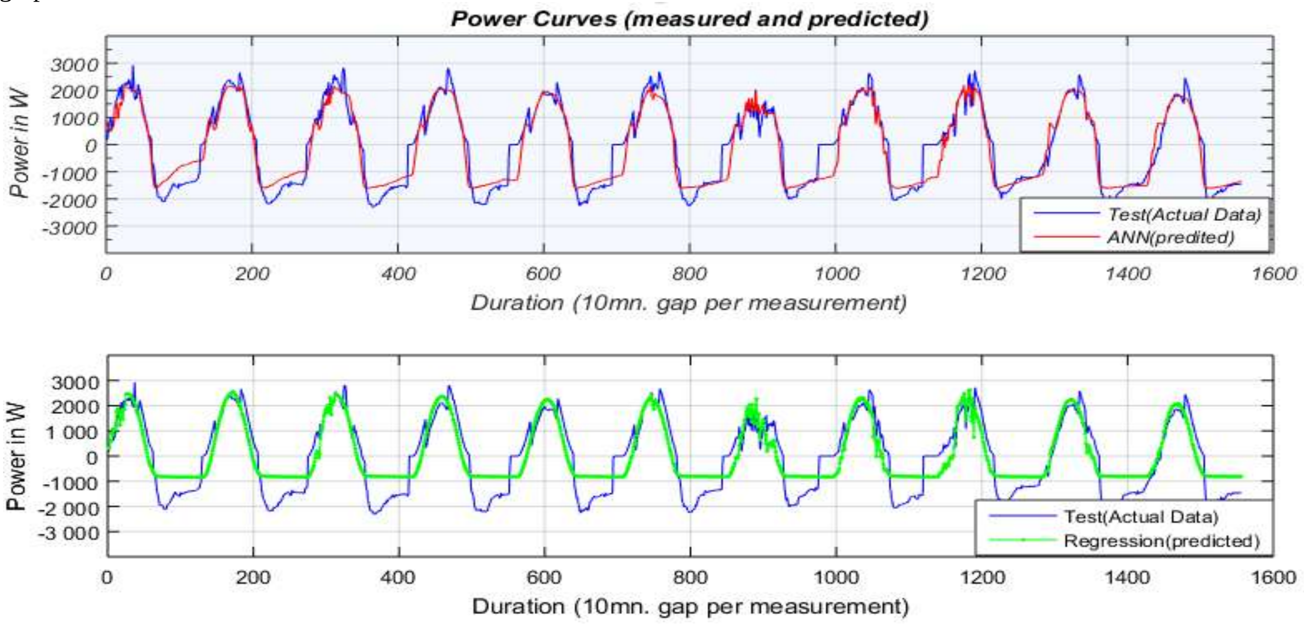


Figure IV-4:- Result of the simulation with cascade-forward architecture (top) compared to regression (bottom).

Tableau IV-2:- Error table (cascade-forward and regression).

	MAE	MAPE (%)	RMSE
ANNs (cascade-forward)	353.15	26.515	472.44
Mutivariableregression	655.11	49.186	738.22

The results of our simulations give us the following summary table:

Tableau IV-3:- Error summary table for all models.

	MAE	MAPE (Accuracy)	RMSE (Efficiency)
ANNs (feed-forward)	326.17	24.489 %	436.08
ANNs (cascade-forward)	353.15	26.515 %	472.44
Multivariableregression	655.11	49.186 %	738.22

We notice that the direct-action neural network (feed-forward) is more efficient than the other models used. Indeed, with an RMSE=436.08 we see that it offers a smaller deviation of the predicted value compared to the measured one. Moreover, we have a better general accuracy with this model (MAPE=24.489%).

This model can now be used to predict the production of a PV installation for 10 days. All that is needed is the weather forecast for the next 10 days.

Ultimately, using this approach, we are able to setup a large panel of prosumers (different meteorological data) and simulate the evolution of their energy excess. This will allow them to negotiate energy purchase contracts.

Conclusion:-

This paper demonstrated the efficiency of artificial neural networks with two different topologies (looped and unlooped) in predicting the production overflow of a prosumer Access to the latest and most recent data can increase the efficiency of the designed models by improving the generalizability of the ANNs used.

Furthermore, the models developed in this paper can be improved by trying other architectures (other activation functions, other learning algorithms ...). One can also try to integrate other input variables and simulate other parameters or use this approach on other PV plant configurations integrating several sources for example.

Références:-

1. G. Guérard, Optimisation de la diffusion de l'énergie dans les smart grids, Université de Versailles - Ecole doctorale sciences et technologies, 2014.
2. N. GENSOLLEN, Modeling and optimizing a distributed power network: A complex system approach of the prosumer management in the smart grid, Paris: École doctorale Informatique, Télécommunications et Électronique (Paris), 2016.
3. Y. Djeriri, «Les réseaux de neurones artificiels,» University of Sidi-Bel-Abbes, 2017.
4. T. Khatib et W. Elmenreich, Modeling of photovoltaic systems using MATLAB®: simplified green codes / by Tamer Khatib, WilfriedElmenreich, N. J.:. J. W. &. S. I. [. |. Hoboken, Éd., New Jersey: John Wiley & Sons, Inc. All rights reserved, 2016.
5. P. N. Ramesh Babu, SMART GRID SYSTEM: Modeling and Control, Waretown, NJ 08758: Apple Academic Press, Inc., 2019.
6. C. Voyant, «Prédiction de séries temporelles de rayonnement solaire global et de production d’énergie,» Université de Corse-Pascal Paoli, Corse, 2011.
7. B. Pillot, Planification de l’électrification rurale décentralisée en Afrique subsaharienne à l’aide de sources renouvelables d'énergie : le cas de l'énergie photovoltaïque en République de Djibouti, Université Pascal Paoli, 2014.
8. R. J. Christophe Pévré, Guide de conception des réseaux électriques indutriels. Schneider, Electric, 1997.
9. P. R. Eric Félice, Qualité des réseaux électriques et efficacité énergétique, Paris: Dunod, 2009.
10. Centre de la CEDEAO pour les Energies Renouvelables et l'Efficacité Energétique (CERBEC), «Politique d'Energies Renouvelables de la CEDEAO,» 2015.